CLAIMS

What is claimed is:

1   1.   A method, comprising:

2   enabling a processor to access first and second memory stores;

3   employing a first portion of the first memory store as a general-purpose

4   memory store;

5   employing a second portion of the first memory store as a memory cache for

6   the second memory store.

1   2.   The method of claim 1, wherein the first memory store comprises an off-chip

2   static random access memory (SRAM) memory store.

1   3.   The method of claim 1, wherein the second memory store comprises an off-

2   chip dynamic random access memory (DRAM) memory store.

1   4.   The method of claim 1, wherein the second memory store comprises an off-

2   chip memory store comprising one of Rambus dynamic random access memory

3   (RDRAM), RLDRAM (reduced latency DRAM), DDR (double data rate), DDR-2, or

4   DDR-3 DRAM, and FCDRAM (fast cycle DRAM).

1   5.   The method of claim 1, further comprising:

2   employing a hardware-based cache management component to manage

3   access to the memory cache.

1   6.   The method of claim 5, further comprising:

2   maintaining a cache tag array implemented in the hardware-based cache

3   management component.

1   7.   The method of claim 5, wherein the hardware-based cache management

2   component includes a content addressable memory (CAM) component, the method

3   further comprising:

4   managing access to the memory cache via the CAM component.

1   8.   The method of claim 1, further comprising:

2   partitioning the memory cache into at least one data cache and at least one

3   address tag array.

1   9.   The method of claim 8, wherein the processor supports concurrent execution

2   of a plurality of threads, the method further comprising:

3   maintaining an address tag array for each of said plurality of threads; and

4   maintaining a data cache for each of said plurality of threads.

1   10.   The method of claim 1, wherein the processor comprises a network

2   processor.

1   11.   The method of claim 10, further comprising:

2   integrating a front-side bus controller on the network processor; and

3   coupling a memory store to the network processor via a front-side bus.

1   12.   The method of claim 10, further comprising:

2   employing the second memory store as a memory store for a general-

3   purpose processor integrated into the network processor.

1   13.   A method, comprising:

2   enabling a plurality of network processors to access first and second shared

3   memory stores;

4   employing a first portion of the first shared memory store as a memory store;

5   employing a second portion of the first shared memory store as a memory

6   cache for the second shared memory store.

1   14.   The method of claim 13, wherein the first shared memory store comprises

2   static random access memory (SRAM).

1   15.   The method of claim 13, wherein the second shared memory store comprises

2   dynamic random access memory (DRAM).

1   16.   The method of claim 13, further comprising:

2   configuring the memory cache as a cache array including a plurality of cache

3   lines; and

4   maintaining a local address-to-cache line map on each of the plurality of

5   network processors in which cache lines associated with that network processor are

6   mapped to corresponding memory addresses in an address space for the second

7   shared memory store.

1   17.   The method of claim 16, further comprising:

2   broadcasting a cache line access request to the plurality of network

3   processors in response to a memory access request from a requesting network

4   processors; and

5   performing a cache line lookup in the local address-to-cache line map of each

6   network processor to determine which, if any, network processors owns a cache line

7   corresponding to the memory access request.

1   18.   The method of claim 17, further comprising:

2   accessing the cache line to service the memory access request if it is

3   determined that one of the network processors owns the cache line; and

4   modifying entries in the local address-to-cache line maps for each of the

5   network processor that owned the cache line and the requesting network processor

6   to reassign ownership of the cache line to the requesting network processor.

1   19.   The method of claim 17, further comprising:

2   accessing memory from the second shared memory store if it is determined

3   that none of the network processors owns the cache line;

4   selecting a cache line in the memory cache to replace;

5   copying data corresponding to data contained in the memory in the second

6   shared memory store that is accessed to the cache line selected to be replaced; and

7   modifying the local address-to-cache line map of the requesting network

8   processor to assign ownership of the cache line that is replaced to the requesting

9   network processor.

1   20.   A network processor, comprising:

2   an internal interconnect;

3   a plurality of packet processing micro-engines, each coupled to the internal

4   interconnect;

5   a static random access memory (SRAM) controller, coupled to the internal

6   interconnect, to control access to an off-chip SRAM memory store;

23

7   a general-purpose processor, coupled to the internal interconnect; and

8   a cache management component, coupled to the internal interconnect, to

9   effectuate a memory cache in a first portion of the off-chip SRAM memory store,

10   wherein a second portion of the off-chip SRAM memory store may be

11   accessed as a non-cached memory store.

1   21.   The network processor of claim 20, further comprising:

2   a front-side bus (FSB) interface; and

3   an FSB controller, coupled between the FSB interface and the internal

4   interconnect,

5   wherein the cache management component is programmed to manage a

6   memory cache in the SRAM memory store corresponding to a memory address

7   space for a dynamic random access memory (DRAM) memory store to be accessed

8   via the FSB interface.

1   22.   The network processor of claim 20, further comprising:

2   a private communication channel linking the SRAM controller to the cache

3   management component.

1   23.   The network processor of claim 20, further comprising:

2   a Rambus dynamic random access memory (RDRAM) interface; and

3   an RDRAM controller, coupled between the RDRAM interface and the internal

4   interconnect,

5   wherein the cache management component is programmed to manage a

6   memory cache in the SRAM memory store corresponding to a memory address

7   space for a RDRAM memory store to be accessed via the RDRAM interface.

1   24.   The network processor of claim 20, wherein the cache management

2   component includes facilities for maintaining a hardware-based cache tag array.

1   25.   The network processor of claim 20, wherein the cache management

2   component includes facilities to manage a cache tag array in a portion of the SRAM

3   memory store.

1   26.   The network processor of claim 20, wherein the cache management

2   component includes facilities to effectuate a content addressable memory (CAM)

3   cache management scheme.

1   27.   A system, comprising:

2   a static random access memory (SRAM) memory store, coupled to an SRAM

3   interface;

4   a dynamic random access memory (DRAM) memory store, coupled to a front-

5   side bus (FSB); and

6   a network processor, including:

7   an internal interconnect;

8   a plurality of packet processing micro-engines, each coupled to the

9   internal interconnect;

10   a static random access memory (SRAM) controller, coupled between

11 the internal interconnect and the SRAM interface;

12   a general-purpose processor, coupled to the internal interconnect;

13   an FSB interface, coupled to the front-side bus;

14   an FSB controller, coupled between the internal interconnect and the

15 FSB interface; and

16   a cache management component, coupled to the internal interconnect,

17   to effectuate a memory cache in a first portion of SRAM memory store,

18   wherein a second portion of the SRAM memory store may be accessed as a

19 non-cached memory store.

1   28.   The system of claim 27, further comprising:

2   a Rambus dynamic random access memory (RDRAM) memory store,

3 coupled to an RDRAM interface; and

4   an (RDRAM) controller, integrated on the network processor and coupled

5 between the RDRAM interface and the internal interconnect.

1   29.   The system of claim 27, wherein the FSB controller and the cache

2 management component are integrated.

1   30.   The system of claim 27, wherein the cache management component includes

2 facilities for maintaining a hardware-based cache tag array.